UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/813,615	03/31/2004	Simon Knowles	66365-020	3818

7590 08/04/2009

MCDERMOTT, WILL & EMERY
600 13th Street, N.W.
Washington, DC 20005-3096

EXAMINER	
FENNEMA, ROBERT E	

ART UNIT	PAPER NUMBER
2183	

MAIL DATE	DELIVERY MODE
08/04/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/813,615	KNOWLES, SIMON	
	Examiner	Art Unit	
	ROBERT E. FENNEMA	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 June 2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18 and 21* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-18 and 21* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-18 and 21 have been considered. Claims 1, 18, and 21 amended as per Applicant's request.

Claim Objections

2. Claim 21 is objected to, as the claim currently reads as a "computer program product", however, this language was rejected under 101 in a previous action, and the Applicant had amended the claim to read as "a computer readable medium". However, Applicant appears to have reverted to the "computer program product" language in the claims filed on 3/10/2008, without indicating the amendment. Examiner did not notice this change at the time, however, believes this to be a typographical error, given the lack of underlining in the amendment filed on 3/10/2008, and the previous amendments (filed on 9/22/2006 and continued on 6/4/2007) changing the claim to a medium. However, if this was designed to be intentional, the 101 rejection on the claim would be reinstated, as a computer program product does not fall into any of the statutory categories of invention. Examiner is interpreting Claim 21's preamble as "A computer-readable medium comprising a sequence of instruction packets" for the remainder of this action, but correction on the Applicant's part is required in the next action to clarify what the claim is supposed to read, but language other than how the Examiner is interpreting the claim will result in a 101 rejection.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-7, 11, 14-18, and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hull, in view of "Unifying FPGAs and SIMD Arrays" by Bolotski et al. (herein Bolotski), further in view of Hennessy et al. ("Computer Architecture: A Quantitative Approach", herein Hennessy).

5. As per Claim 1, Hull teaches: A computer processor for processing (i) instruction packets comprising a plurality of only control instructions (Figure 4, Template "B"), the control instructions having a control bit width (inherent), and (ii) instruction packets comprising a plurality of instructions comprising at least one data processing instruction (Figure 4, for example, Template "E, the processor comprising:

a decode unit for decoding sequentially the instruction packets fetched from a memory holding a sequence of instruction packets (Column 5, Lines 16-18);

a control processing channel capable of performing control operations, the control processing channel comprising a plurality of functional units (Column 3, Line 51, the branch execution unit) including a control register file having a first bit width (Column 3, Lines 18-21, 64 bits); and

a data processing channel capable of performing data processing operation, the data processing channel comprising a plurality of functional units (Column 3, Lines 48-

50, the integer, memory, and floating point execution units) including a data register file having a second bit width wider than the first bit width (Column 3, Lines 9-11, the floating point registers have a bit width of 82 bits);

wherein the decode unit comprises decode circuitry configured to decode identification bits of each instruction packet to determine which type (i), (ii) of instruction packet is being decoded, and control circuitry configured to pass the plurality of only control instructions from an instruction packet of type (i) to the control processing channel when the decode circuitry indicates so and to pass the plurality of instructions comprising at least one data processing instruction from an instruction packet of type (ii) to the data processing channel when the decode circuitry indicates so (Column 3, Lines 63-66, the template field in each packet/bundle tells the machine what is in each packet, and directs the packet to the appropriate functional units. Template "B" for example contains 3 branch (control) instructions, and Template "0" contains only data-processing instructions);

wherein, in use the decode unit causes instructions of (i) instruction packets comprising a plurality of only control instructions to be executed sequentially on the control processing channel (Column 4, Lines 61-62); and

wherein in use the decode unit causes instructions of (ii) instruction packets comprising a plurality of instructions comprising at least one data processing instruction to be executed simultaneously on the data processing channel (Column 2, Lines 5-9), but fails to teach:

at least one input of which is a vector, and

the data processing instructions having a data processing bit width wider than the control bit width.

Hull is silent towards vectors (using the Applicant's definition of a vector in Paragraph 23 of the specification, wherein a vector is an assembly of scalar operands). Hull does however teach that there is a high demand for highly efficient, parallel processing machines, and Bolotski teaches that a massively parallel system of computing is a SIMD processor, which is a vector, or array, of scalar values. Given the disclosed need in Hull of increased parallelism, the teachings of both Bolostki of the parallelism of SIMD (along with SIMD being well known in the art), one of ordinary skill in the art would have been motivated to integrate SIMD processing into the system of Hull, to further exploit parallelism in lines with the goals of Hull.

However, Hull and Bolostki are silent towards the data processing instructions being wider (longer) than a control instruction, because Hull teaches a fixed-length architecture. However, Hull does note in Column 1, Lines 15-23 that previous instruction sets used variable-length architecture, but Hull happens to be designed for a RISC machine. Hennessy (a textbook) teaches the differences between a variable and fixed architecture, and shows that variable-length instructions create smaller programs (Page 128), and that on average, the instructions are significantly smaller than RISC instructions (Page 128 and Page 129, Figure 2.23) because they do not require wasted and unneeded fields. Given that Hull's invention is designed to reduce waste and inefficiency in code size (Column 2, Lines 3-5), and to help avoid the expanded code caused by fixed-length instructions (Column 1, Lines 52-60), one of ordinary skill in the

art would have been motivated to implement Hull's invention in a variable-length instruction set, to further increase the flexibility of the code, and reduce its size, while still taking advantage of the alignment and efficiencies provided by Hull's templates.

Given that combination, Examiner notes Page D-13 of Hennessy, which shows that the control instruction (a, JE), has a smaller length than at least two data processing instructions (c, MOV and e, ADD) in these examples.

6. As per Claim 2, Hull teaches: A computer processor according to claim 1, wherein the control processing channel further comprises a branch unit and a control execution unit (Colum

7. As per Claim 3, Hull teaches The computer processor according to claim 1, wherein the data processing channel further comprises a fixed data execution unit (Column 3, Lines 48-51), but fails to teach:

the second processing channel containing a configurable data execution unit.

While Hull teaches a fixed data execution unit, Hull is silent towards a configurable data execution unit. However, Bolotski teaches a system that can simulate SIMD and configurable operations on the same unit (Section 4), which can be subdivided until SIMD and configurable units. Boloski further teaches that the advantages of combining a SIMD and configurable unit includes reducing cost by not duplicating logic (Section 4.1), additionally, one of ordinary skill in the art would recognize the additional benefits of configurable hardware, such as being able to

configure exactly the units you need, allowing for flexibility and increased performance. Therefore, one of ordinary skill in the art would have been motivated to combine the teachings of Hull and Bolotski to implement configurable data execution units.

8. As per Claim 4, Bolotski teaches: A computer processor according to claim 3, wherein the fixed data execution unit and the configurable data execution unit both operate according to a single instruction multiple data format (Section 4).

9. As per Claim 5, Hull teaches: The computer processor according to claim 1, wherein the control and data processing channels share a load store unit (Column 3, Line 50, there is only one memory execution unit, and since the branch unit needs to be able to have access to memory (to fetch the new target instructions), it clearly has to be shared between the two).

10. As per Claim 6, Hull teaches: A computer processor according to claim 5, wherein the load store unit uses control information supplied by the control processing channel and data supplied by the data processing channel (Column 3, Line 50, there is only one memory execution unit, the data processing channel needs to use it to fetch data from memory, and the control channel needs to use it to fetch branch target addresses).

11. As per Claim 7, Hull teaches: A computer processor according to claim 1, wherein the instruction packets are all of equal bit length (Column 2, Lines 22-24, 128 bit bundles).

12. As per Claim 11, Hull teaches: A computer processor according to claim 7, wherein the nature of each instruction in an instruction packet is selected at least from a control instruction, a data instruction, and a memory access instruction (Column 3, Lines 48-51, also see Figure 4).

13. As per Claim 14, Hull teaches: A computer processor according to claim 1, wherein when the decode unit detects that the instruction packet defines three control instructions, the decode unit is operable to supply the control processing channel with the three control instructions (Column 3, Lines 65-66) whereby the three control instructions are executed sequentially (Column 4, Lines 61-62).

14. As per Claim 15, Hull teaches: A computer processor according to claim 1, wherein when the decode unit detects that the instruction packet defines two instructions comprising at least one data instruction, the decode unit is operable to supply the second processing channel with at least the data instruction (Column 3, Lines 65-66) whereby the two instructions are executed simultaneously (Column 2, Lines 5-9).

15. As per Claim 16, Hull teaches: A computer processor according to claim 1, wherein the decode unit is operable to read the values of a set of designated bits at predetermined bit locations in each instruction packet of the sequence (Column 3, Lines 63-66, the template bits), to determine:

a) whether the instruction packet defines a plurality of only control instructions (Figure 4, Template "B") or a plurality of instructions of which at least one is a data instruction (Figure 4, any non-"B" template); and

b) where the instruction packet defines a plurality of instructions of which at least one is a data instruction, the nature of each of the two instructions selected from: a control instruction; a data instruction; and a memory access instruction (Figure 4, see Template "0" or "8" for example).

16. As per Claim 17, Hull teaches: A computer processor according to claim 3, wherein the configurable data execution unit is capable of executing more than two consecutive operations on the data provided by a single issued instruction before returning a result to a destination register file.

One of ordinary skill in the pertinent art would have recognized that this is a simple accumulate function that would be easily programmable in configurable logic.

17. As per Claim 18, Hull teaches: A method of operating a computer processor for processing (i) instruction packets comprising a plurality of only control instructions (Figure 4, Template "B"), the control instructions having a control bit width (inherent);

and (ii) instruction packets comprising a plurality of instructions comprising at least one data processing instruction (Figure 4, Template "0"),

the processor comprising a decode unit for decoding sequentially the instruction packets fetched from a memory holding the instruction packets (Column 5, Lines 16-18):

a control processing channels comprising a plurality of functional units (Column 3, Line 51, the branch execution unit) including a control register file having a first bit width (Column 3, Lines 18-21, 64 bits) and a data processing channel comprising a plurality of functional units including a data register file having a second bit width, wider than the first bit width (Column 3, Lines 9-11, the floating point register file has a bit width of 82 bits), the method comprising:

decoding identification bits of each instruction packet to determine which type (i), (ii), of instruction packet is being decoded, and passing the plurality of only control instructions from an instruction packet of type (i) to the control processing channel when the decode circuitry indicates so and passing the plurality of instructions comprising at least one data processing instruction from an instruction packet of type (ii) to the data processing channel when the decode circuitry indicates so (Column 3, Lines 63-66, the template field in each packet/bundle tells the machine what is in each packet, and directs the packet to the appropriate functional units. Template "B" for example contains 3 branch (control) instructions, and Template "0" contains only data-processing instructions);

when the instruction packet defines (i) a plurality of only control instructions, supplying the control instructions to the control processing channel wherein the control instructions are executed sequentially (Column 4, Lines 61-62); and

when the instruction packet defines (ii) a plurality of instructions comprising at least one data processing instruction, supplying at least the data instruction to the data processing channel wherein the plurality of instructions are executed simultaneously (Column 2, Lines 5-9), but fails to teach:

the data processing instructions having a data processing bit width wider than the control bit width; and

a data processing channel capable of performing data processing operations at least one input of which is a vector.

Hull is silent towards vectors (using the Applicant's definition of a vector in Paragraph 23 of the specification, wherein a vector is an assembly of scalar operands). Hull does however teach that there is a high demand for highly efficient, parallel processing machines, and Bolotski teaches that a massively parallel system of computing is a SIMD processor, which is a vector, or array, of scalar values. Given the disclosed need in Hull of increased parallelism, the teachings of both Bolostki of the parallelism of SIMD (along with SIMD being well known in the art), one of ordinary skill in the art would have been motivated to integrate SIMD processing into the system of Hull, to further exploit parallelism in lines with the goals of Hull.

However, Hull and Bolostki are silent towards the data processing instructions being wider (longer) than a control instruction, because Hull teaches a fixed-length

architecture. However, Hull does note in Column 1, Lines 15-23 that previous instruction sets used variable-length architecture, but Hull happens to be designed for a RISC machine. Hennessy (a textbook) teaches the differences between a variable and fixed architecture, and shows that variable-length instructions create smaller programs (Page 128), and that on average, the instructions are significantly smaller than RISC instructions (Page 128 and Page 129, Figure 2.23) because they do not require wasted and unneeded fields. Given that Hull's invention is designed to reduce waste and inefficiency in code size (Column 2, Lines 3-5), and to help avoid the expanded code caused by fixed-length instructions (Column 1, Lines 52-60), one of ordinary skill in the art would have been motivated to implement Hull's invention in a variable-length instruction set, to further increase the flexibility of the code, and reduce its size, while still taking advantage of the alignment and efficiencies provided by Hull's templates.

Given that combination, Examiner notes Page D-13 of Hennessy, which shows that the control instruction (a, JE), has a smaller length than at least two data processing instructions (c, MOV and e, ADD) in these examples.

18. As per Claim 21, Hull teaches: A computer-readable medium comprising a sequence of instruction packets (Column 2, Lines 22-24),

said instruction packets including a first type of instruction packet comprising a plurality of only control instructions of equal width (Figure 4, Template "B"), the control instructions having a control bit width (inherent), and a second type of instruction packet comprising a plurality of instructions including at least one data processing instruction

(Figure 4, Template "0"), said instruction packets including at least one indicator bit at a designated bit location within the instruction packet (Column 3, Lines 63-66, the template fields), wherein the computer-readable medium is adapted to run on a computer such that said indication bit is adapted to cooperate with a decode unit of the computer to designate whether:

a) the instruction packet defines a plurality of only control instructions or a plurality of instructions comprising at least one is a data instruction (Column 3, Lines 63-66); and

b) in the case when there is a plurality of instructions comprising at least one data instruction, the nature of each of the first and second instructions selected from: a control instruction; a data instruction; and a memory access instruction (Figure 4, see Templates "0" or "8" for example), but fails to teach:

the at least one data processing instructions having a data processing bit width wider than the control bit width, and wherein at least one data processing instruction is a vector.

Hull is silent towards vectors (using the Applicant's definition of a vector in Paragraph 23 of the specification, wherein a vector is an assembly of scalar operands). Hull does however teach that there is a high demand for highly efficient, parallel processing machines, and Bolotski teaches that a massively parallel system of computing is a SIMD processor, which is a vector, or array, of scalar values. Given the disclosed need in Hull of increased parallelism, the teachings of both Bolostki of the parallelism of SIMD (along with SIMD being well known in the art), one of ordinary skill

in the art would have been motivated to integrate SIMD processing into the system of Hull, to further exploit parallelism in lines with the goals of Hull.

However, Hull and Bolostki are silent towards the data processing instructions being wider (longer) than a control instruction, because Hull teaches a fixed-length architecture. However, Hull does note in Column 1, Lines 15-23 that previous instruction sets used variable-length architecture, but Hull happens to be designed for a RISC machine. Hennessy (a textbook) teaches the differences between a variable and fixed architecture, and shows that variable-length instructions create smaller programs (Page 128), and that on average, the instructions are significantly smaller than RISC instructions (Page 128 and Page 129, Figure 2.23) because they do not require wasted and unneeded fields. Given that Hull's invention is designed to reduce waste and inefficiency in code size (Column 2, Lines 3-5), and to help avoid the expanded code caused by fixed-length instructions (Column 1, Lines 52-60), one of ordinary skill in the art would have been motivated to implement Hull's invention in a variable-length instruction set, to further increase the flexibility of the code, and reduce its size, while still taking advantage of the alignment and efficiencies provided by Hull's templates.

Given that combination, Examiner notes Page D-13 of Hennessy, which shows that the control instruction (a, JE), has a smaller length than at least two data processing instructions (c, MOV and e, ADD) in these examples.

19. Claims 8-10, 12 and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hull, Hennessy, and Bolotski, in view of In re Rose.

20. These claims recite limitations of the bit lengths of various instructions and packets.

While Hull may not teach the recited lengths, if it was advantageous to lengthen or shorten the bit lengths in Hull for various reasons, one of ordinary skill in the pertinent art would have recognized that it would have been simple to do so.

Further, it has been found that a change in size does not produce a patentable distinction. In re Rose , 220 F.2d 459, 105 USPQ 237 (CCPA 1955) (Claims directed to a lumber package "of appreciable size and weight requiring handling by a lift truck" where held unpatentable over prior art lumber packages which could be lifted by hand because limitations relating to the size of the package were not sufficient to patentably distinguish over the prior art.); In re Rinehart, 531 F.2d 1048, 189 USPQ 143 (CCPA 1976) ("mere scaling up of a prior art process capable of being scaled up, if such were the case, would not establish patentability in a claim to an old process so scaled." 531 F.2d at 1053, 189 USPQ at 148.).

Response to Arguments

21. In regards to Applicant's arguments that Hull does not teach a variable-length instruction set, and thus, cannot teach a control instruction with a bit-width shorter than a data processing instruction, Examiner finds the arguments persuasive. However, Examiner has provided a new grounds of rejection including Hennessy.

Hennessy is a textbook which shows that there are three ways to implement an instruction set, with variable length, with fixed length, or with a hybrid of the two. Each way has advantages and disadvantages, and Examiner asserts that it would have been obvious to switch from one to the other, depending upon whether complexity or code size was more important. Hull appears to indicate that his invention provides several advantages over the prior art, which coincides with the advantages of a variable-length instruction set. Given that Hull's invention is not exclusive to a fixed-length instruction set (it is designed for one, but there is no reason why it cannot be used in a variable-length instruction set, as it can provide the same advantages), Examiner believes one of ordinary skill in the art would have been motivated to implement Hull's invention in a variable-length environment, and that taking an invention, and implementing it in either a variable-length, or fixed-length system, by itself, is not a patentable distinction for the reasons laid out in Hennessy, and that a further distinction must be made over the prior art for the current rejection to be overcome.

22. Applicant has argued on Page 10 of the remarks that in the current invention, a packet can contain either two instructions, or three instructions, and that Hull always has three instructions. However, this feature is not found in the claims, thus Examiner is not persuaded by this argument. Even if it were explicitly recited in the claims, Examiner would need to be convinced that it would not be an obvious modification when switching from fixed-length to variable-length instructions.

23. Applicant has also argued on Page 11 that an allegedly distinguishing feature in the claims is that control instructions are executed sequentially, and that data processing instructions must process the instructions simultaneously. Applicant has supported this argument by stating that the stop bits are not present in template B, thus that the branch instructions could be executed in any order. However, Examiner disagrees, as this would fundamentally change the way the processor would work. Given that they are control transfer instructions, if one was to execute, for example, the second branch, when the first one was taken, the path of execution would be incorrect. Examiner does not believe it is reasonable to interpret Hull in such a way that would lead to incorrect operation. Hull states that execution order proceeds from slot 0 to slot 2 (Column 4, Lines 61-62), the stop bits are used to separate instruction groups (and to allow for dependency), not to force sequential operation (although that is a by-product). Additionally, Examiner sees no evidence that there is more than one branch execution unit (as having more would be useless, as executing branches out of order would be disastrous), thus, it would be impossible to execute all three branches in Hull simultaneously, even if one ignored the common sense that would prevent one from doing so. Therefore, while Hull teaches that instructions in the same bundle are generally processed simultaneously, Examiner believes one must make common sense interpretations, and recognize that for certain templates, such as B, that the instructions cannot possibly be executed simultaneously. Additionally, it would appear that the stop bits as shown in Figure 4 are added by the compiler, and are not a part of the template itself, and that the shown stop bits are for example purposes, as they are to show

dependencies, which could not possibly be taken into account without the code. Therefore, Examiner believes there is ample evidence, both given Hull's explicit disclosure, and common sense and knowledge in the art, to show that control instructions execute in sequence.

Conclusion

24. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to ROBERT E. FENNEMA whose telephone number is (571) 272-2748. The examiner can normally be reached on Monday-Thursday, 9:30-6:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Eddie P Chan/
Supervisory Patent Examiner, Art Unit 2183

Robert E Fennema
Examiner
Art Unit 2183

RF